上級編

第 40 講　中央値の判定

例題

以下の条件で，中央値を求めるプログラムを作成する。

- 例えば, 昇順に並べられた配列 1,3,6,7,9 の場合の中央値はちょうど真ん中の 6 となる。
- 配列 1,3,7,9 の場合は，ちょうど真ん中は 3 と 7 の間なので 3 と 7 の平均を求めて 5 となる。
- 「//」は，商の整数値を求める演算子である。
- 「%」は，余りを求める演算子である。

以下のプログラムの空欄に入る最も適切なものを，あとの①～④のうちから一つ選びなさい。

```
Data = [1, 3, 7, 9] # 昇順のデータ配列
center = 0 # 算出した中央値を格納
n = len(Data) # n に要素数を格納

if n % 2 == 0:
  center = (Data[n // 2 - 1] + Data[n // 2]) / 2
else:
  center = [        ]

print(' 中央値：', int(center))
```

① `Data[n // 2]`

② `Data[n % 2]`

③ `Data[n // 2] / 2`

④ `Data[n // 2 + 1] + Data[n // 2]`

＜MEMO＞

上級編　第 40 講　確認問題

以下の条件で，第１四分位数を求めるプログラムを作成する。

・第１四分位数とは，データを小さい順に並べたとき，初めから数えて 25% の位置にある数のことである。

・例えば，昇順に並べられたデータ数が偶数個の配列 [1, 3, 6, 7, 8, 9] の場合は，1,3,6 と 7,8,9 の２つのグループに分け，1,3,6 の真ん中の３が第１四分位数になる。

・データ数が奇数個の配列 [1,4,6,8,9,10,13] の場合は，真ん中の８を除いて 1,4,6 と 9,10,13 の２つのグループに分け，1,4,6 の真ん中の４が第１四分位数になる。

・左側のグループのデータが 1,5,7,9 のように偶数個の場合は，５と７の平均をとって６が第１四分位数となる。

以下のプログラムの空欄に入る最も適切なものを，あとの①～④のうちから一つ選びなさい。

```
Data = [1, 4, 6, 8, 9, 10, 13, 15] # 昇順のデータ配列
Left_data = [] # 左側のグループのデータを入れる配列
n = len(Data)
midpoint = n // 2

# for 文を使用して，左側のグループデータを作成
for i in range(0, midpoint):
  Left_data.append(Data[i])

left_n = len(Left_data)

# 左側のデータセットから中央値を取得
if left_n % 2 == 0:
  first_quartile = □
else:
  first_quartile = Left_data[left_n // 2]

print(int(first_quartile))
```

① Left_data[left_n // 2 + 1] + Left_data[left_n // 2]

② Left_data[left_n // 2] / 2

③ (Left_data[left_n // 2 - 1] + Left_data[left_n // 2]) / 2

④ Left_data[left_n - 1] + Left_data[left_n]

上級編　例題　解答

第34講　コインの枚数

例題　解答：④

第35講　駐車料金の判定

例題　解答：④

第36講　割引のある料金

例題　解答：④

第37講　テストの集計

例題　解答：①

第38講　じゃんけん

例題　解答：①

第39講　サイコロの出目

例題　解答：③

第40講　中央値の判定

例題　解答：①

第41講　素数判定

例題　解答：③

第42講　素因数分解

例題　解答：②

第43講　倍数の判定・FizzBuzz

例題　解答：④

第44講　バブルソート・選択ソート

例題　解答：③

第45講　グラフ理論

例題　解答：②

第46講　待ち行列

例題　解答：②

第47講　パリティチェック

例題　解答：①

第48講　すごろくゲーム

例題　解答：④

上級編　第 40 講　確認問題　解答

正解：③

解説：

左側のグループのデータ数が，奇数個のときと偶数個のときで，処理を分ける必要がある。

13行目の`if left_n % 2 == 0`では，左側のグループのデータ数が偶数個であるかを判定している。よって，この条件式が True であるとき，問題文の条件４つ目のとおり，左側のグループデータを真ん中で分割したときの，前後２つの数の平均を求めればよい。 そのため，14 行目には，「グループデータの要素数を真ん中で分割したときの，前の数（真ん中の前の数）と，その次の数（真ん中の後ろの数）を足して，2 で割る式」が当てはまると考えられる。よって，正解は③ `(Left_data[left_n // 2 - 1] + Left_data[left_n // 2]) / 2` である。

```
01 Data = [1, 4, 6, 8, 9, 10, 13, 15] # 昇順のデータ配列
02 Left_data = [] # 左側のグループのデータを入れる配列
03 n = len(Data)
04 midpoint = n // 2
05
06 # for 文を使用して，左側のグループデータを作成
07 for i in range(0, midpoint):
08   Left_data.append(Data[i])
09
10 left_n = len(Left_data)
11
12 # 左側のデータセットから中央値を取得
13 if left_n % 2 == 0:
14   first_quartile = (Left_data[left_n // 2 - 1] + Left_data[left_n // 2]) / 2
15 else:
16   first_quartile = Left_data[left_n // 2]
17
18 print(int(first_quartile))
```

実行結果

```
5
```